IN THE CLAIMS

Please cancel claim 5 without prejudice or disclaimer and amend claims 1, 2, 4, 6 and 9 as follows:

1. (Currently amended) A process for restricting unauthorized operations by a computer user in a multi-user system, comprising the steps of:

automatically using a security executable to create a list of authorized operations for said computer user when the computer user logs on to the multi-user system;

attaching a hook function to all new processes;

employing the hook function whenever a new application is started to send a message to the security executable, said message including the process ~~[id]~~ ID and path of the new application;

receiving said message from the hook function at the security executable and correlating to said list to determine whether the new application is authorized ~~or not~~;

answering the message by the security executable when the new application is authorized ~~to indicate so~~ and;

stopping the new application when the new application is not authorized.

2. (Currently amended) A software system for restricting unauthorized operations by a computer user in a multi-user system, comprising:

a first program module~~, which is~~ comprising a hook procedure[,] for automatically attaching to all new processes when the computer user logs on to the multi-user system and for querying an ID of each said new process; and

a second program module in communication with said first program module, said second program module using a security executable to build a list of allowed applications, retrieve the ID of each new process from said first program module, and terminate each new process not identified on said list of allowed applications.

3. (Original) The software system for restricting unauthorized operations by a computer user according to claim 2, wherein said first program module is executable in user mode.

4. (Currently amended) The software system for restricting unauthorized operations by a computer user according to claim 2, wherein said first program module is attached to said new processes by using [~~the~~] a system dynamic link library.

5. Cancelled.

6. (Currently amended) The software system for restricting unauthorized operations by a computer user according to claim [~~5~~] 4, wherein said first program module communicates with said second program module by sending a message with the process ID and the path of the process being examined.

7. (Original) The software system for restricting unauthorized operations by a computer user according to claim 6, wherein said second program module communicates with said first program module when said process is authorized by answering said message with an indication that said process is authorized.

8. (Original) The software system for restricting unauthorized operations by a computer user according to claim 6, wherein said second program module automatically terminates said process when not authorized.

9. (Currently amended) A process for restricting unauthorized operations by computer users in a network environment, comprising the steps of:

using a security executable to create and maintain a list of authorized processes and IDs for each computer user when the computer user logs on to the network;

attaching a hook function to all new processes;

monitoring all new processes that are started with the hook function and determining [~~an~~] a process ID thereof;

receiving said process ID from the hook function by the security executable;

determining whether the process ID of each started process is on said list;

allowing said process to continue when its process ID is on the list; and

terminating said process when its process ID is not on the list.